

EXPRESS MAIL LABEL NO.: EV 306258343 US

Attorney Docket No.: N1085-00090  
[TSMC2002-1000]

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In Re Patent Application of: Hwa Shin Huang

Serial No.: not yet known

Group Art Unit: To Be Assigned

Filed: 09/15/2003

Examiner: To Be Assigned

For: **SYSTEM AND METHOD FOR DEFINING INTERFACE OF MANUFACTURE  
EXECUTION SYSTEM**

I hereby certify that this correspondence is being deposited  
with the United States Postal Service on the date shown below  
with sufficient postage as "Express Mail Post Office to  
Addressee" using Express Mail Label No **EV 306258343 US**  
under 37 CFR 1.10 addressed to: Mail Stop Patent  
Application, Commissioner for Patents, P.O. Box 1450,  
Alexandria, VA 22313-1450

\_\_\_\_\_  
Date

\_\_\_\_\_  
Joseph A. Powers

Sir:

**TRANSMITTAL OF PRIORITY DOCUMENT**

Attached please find a certified copy of the foreign application from which priority is  
claimed for this case:

<u>Country</u>	<u>Application No.</u>	<u>Filing Date</u>
Republic of China (Taiwan)	092106107	03/19/2003

Respectfully submitted,

Dated: September 15, 2003

Joseph A. Powers  
Joseph A. Powers, Esquire  
Registration No. 47,006  
Attorney for Applicant

DUANE MORRIS LLP  
One Liberty Place  
Philadelphia, Pennsylvania 19103-7396  
(215) 979-1842 (Telephone)  
(215) 979-1020 (Fax)

PH1\1127254.1



中華民國經濟部智慧財產局

INTELLECTUAL PROPERTY OFFICE  
MINISTRY OF ECONOMIC AFFAIRS  
REPUBLIC OF CHINA

茲證明所附文件，係本局存檔中原申請案的副本，正確無訛，  
其申請資料如下：

This is to certify that annexed is a true copy from the records of this  
office of the application as originally filed which is identified hereunder：

申請日：西元 2003 年 03 月 19 日  
Application Date

申請案號：092106107  
Application No.

申請人：台灣積體電路製造股份有限公司  
Applicant(s)

局長  
Director General

蔡練生

發文日期：西元 2003 年 8 月 13 日  
Issue Date

發文字號：09220815610  
Serial No.

申請日期：	IPC分類
申請案號：	

(以上各欄由本局填註)

## 發明專利說明書

一、 發明名稱	中文	定義製造執行系統之介面的系統與方法
	英文	SYSTEM AND METHOD FOR DEFINING INTERFACE OF MANUFACTURE EXECUTION SYSTEM
二、 發明人 (共1人)	姓名 (中文)	1. 黃華昕
	姓名 (英文)	1. HUANG, Hwa-Shin
	國籍 (中英文)	1. 中華民國 TW
	住居所 (中文)	1. 桃園縣桃園市新埔里天祥5街69號12樓
	住居所 (英文)	1. 12F, No. 69, Tien Shiang 5th St., Hsin Pu Li, Taoyuan Hsien
三、 申請人 (共1人)	名稱或姓名 (中文)	1. 台灣積體電路製造股份有限公司
	名稱或姓名 (英文)	1. TAIWAN SEMICONDUCTOR MANUFACTURING CO., LTD.
	國籍 (中英文)	1. 中華民國 TW
	住居所 (營業所) (中文)	1. 新竹科學工業園區新竹縣園區三路121號 (本地址與前向貴局申請者相同)
	住居所 (營業所) (英文)	1. NO. 121, PARK AVE. III, SCIENCE-BASED INDUSTRIAL PARK, HSINCHU, TAIWAN, R. O. C.
	代表人 (中文)	1. 張 忠 謀
	代表人 (英文)	1. Chang, Chung-Mou



四、中文發明摘要 (發明名稱：定義製造執行系統之介面的系統與方法)

一種定義製造執行系統(Manufacturing Execution System ; MES)之介面(Interface)的系統與方法。本發明使用可延伸標記語言(Extensible Markup Language ; XML)形成XML標示設定(Tag-Set)檔和介面定義檔，以簡化製造執行系統SiView所使用介面定義語言(Interface Definition Language ; IDL)檔，並去除介面儲存裝置(Interface Repository ; IFR)，使伺服器(Server)端和客戶(Client)端各只需維護一個介面定義檔和一個XML標示設定檔。本發明更使用XML Schema (綱要)檔來驗證(Validate)XML輸出檔的內容。

伍、(一)、本案代表圖為：第\_\_\_2\_\_\_圖

(二)、本案代表圖之元件代表符號簡單說明：

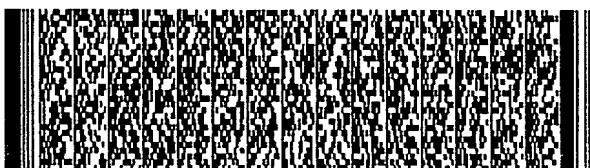
200：伺服器

202：介面定義檔

204：XML標示設定檔

陸、英文發明摘要 (發明名稱：SYSTEM AND METHOD FOR DEFINING INTERFACE OF MANUFACTURE EXECUTION SYSTEM)

A system and a method for defining the interface of a manufacturing execution system (MES) are disclosed. XML (Extensible Markup Language) is used to form an interface definition file and a XML tag-set file for simplifying the IDL (Interface Definition Language) files used by SiView MES and getting rid of IFR (Interface Repositories), so that each of the server and client ends only needs



四、中文發明摘要 (發明名稱：定義製造執行系統之介面的系統與方法)

206 : XML 輸出檔  
210 : 通訊規範  
220 : 客戶端  
230 : 網路伺服器  
232 : XML Schema 檔

陸、英文發明摘要 (發明名稱：SYSTEM AND METHOD FOR DEFINING INTERFACE OF MANUFACTURE EXECUTION SYSTEM)

to maintain a XML tag-set file and an interface definition file. Moreover, a XML schema file is used for validating the contents of the XML output file.



一、本案已向

國家(地區)申請專利

申請日期

案號

主張專利法第二十四條第一項優先權

二、☐主張專利法第二十五條之一第一項優先權：

申請案號：

日期：

三、主張本案係符合專利法第二十條第一項☐第一款但書或☐第二款但書規定之期間

日期：

四、☐有關微生物已寄存於國外：

寄存國家：

寄存機構：

寄存日期：

寄存號碼：

☐有關微生物已寄存於國內(本局所指定之寄存機構)：

寄存機構：

寄存日期：

寄存號碼：

☐熟習該項技術者易於獲得，不須寄存。



## 五、發明說明 (1)

### 【發明所屬之技術領域】

本發明係有關於一種定義製造執行系統(Manufacturing Execution System ; MES)之介面(Interface)的系統與方法，特別有關於一種可延伸標記語言(Extensible Markup Language ; XML)，來簡化SiView製造執行系統的介面定義語言(Interface Definition Language ; IDL)檔的系統與方法。

### 【先前技術】

SiView是美國國際商用機器公司(IBM)所發展出來之新一代的製造執行系統。其係使用美國半導體技術聯盟(SEMATECH)所推薦的共用物件請求仲介者架構(Common Object Request Broker Architecture ; CORBA)為其電腦整合製造系統(Computer Integrated Manufacturing ; CIM)的通訊基礎架構。於CORBA中，用來描述各元件間介面定義的介面定義語言(Interface Definition Language, IDL)扮演者相當重要的角色。

請參照第1圖，其繪示習知之定義製造執行系統之介面的系統架構示意圖。當具有SiView製造執行系統之伺服器(Server)100與客戶(Client)端120以CORBA 110為通訊基礎架構時，必須先編寫IDL檔102，伺服器100與客戶端120的所有提供服務(Services)與請求服務的動作均係透過IDL檔102來執行，其中伺服器100中更須安裝有對各種不同之程式語言(如Visual Basic、C++和Java等)特別設計的複數個介面儲存裝置(Interface Repositories ; IFR)104。編寫



## 五、發明說明 (2)

IDL 檔102 時必須確定有那些物件及這些物件的名稱；這些物件的用途及具備那些功能；物件中需要有那些變數；以及物件之輸入變數為何、輸出變數為何和其例外 (Exception)。IDL 檔102 可以相當簡單或非常複雜，端視整個軟體架構如何設計而定。到目前為止，隨著SiView製造執行系統之運轉時間的增加，SiView中的IDL檔在數量上增加非常多，而且其內容愈來愈複雜，再加上多個介面儲存裝置 104，系統人員必須同時維護相當多而複雜的IDL檔 102 與介面儲存裝置 104，使得管理這些檔案之編碼與部署的經常費用(Overhead)迅速增加，因而耗費許多的人力與物力。另一方面，由執行IDL檔而得之SiView製造執行系統的資料，其格式(Format)係被IDL檔102中的規定所限制，使用者難以更動，故在知識管理上相當不便。而且，SiView製造執行系統缺乏驗證(Validate)的機制來驗證其資料，在知識管理更是缺乏效率。

因此，非常需要發展出一種定義製造執行系統之介面的系統與方法，藉以簡化IDL檔的數目與內容，而大幅地減少管理檔案之編碼與部署的經常費用；使製造執行系統之資料的格式不被限制，而可為有組織的標準化格式；提供驗證製造執行系統之資料的功能，以增加知識管理的效率。

### 【發明內容】

本發明的目的就是在提供一種定義製造執行系統之介面的系統與方法，藉以減少IDL檔的數目至1個，並簡化IDL檔的內容，來大幅地減少管理檔案之編碼與部署的經常費用。





### 五、發明說明 (3)

本發明的又一目的就是在提供一種定義製造執行系統之介面的系統與方法，藉以使製造執行系統之資料的格式不被限制，而可為有組織的標準化格式。

本發明的又一目的就是在提供一種定義製造執行系統之介面的系統與方法，藉以適用於多種通訊規範，而具有可移植性(Portability)。

本發明的再一目的就是在提供一種定義製造執行系統之介面的系統與方法，藉以提供驗證製造執行系統之資料的功能，而增加知識管理的效率

根據本發明之上述目的，提出一種定義製造執行系統之介面的系統，藉以處理伺服器 and 客戶端間的互動

(Transaction)，此伺服器和此客戶端係以可延伸標記語言為基礎(Extensible Markup Language Base ; XML Base)，此伺服器和此客戶端間的互動係建構在通訊規範上，此伺服器具有製造執行系統。

依照本發明一較佳實施例，此判斷造成半導體機台異常之原因的系統至少包括：介面定義檔，用以執行製造執行系統的複數個服務物件(Service Objects)；XML標示設定(Tag Set)檔，其中XML標示設定檔係使用XML來描述服務物件所具有的介面；以及XML Schema(綱要)檔，其中XML Schema檔係建立在網路伺服器(Web server)上，用以驗證(Validate)由執行介面定義檔和XML標示設定檔而得之XML輸出檔的內容，其中XML標示設定檔係用來做為介面定義檔中所描述之服務物件的至少一輸入參數(Argument)。



#### 五、發明說明 (4)

另外，根據本發明之上述目的，提出一種定義製造執行系統之界面的方法，藉以處理伺服器 and 客戶端間的互動，伺服器和客戶端間的互動係建構在通訊規範上，伺服器和客戶端係以XML為基礎(Base)，伺服器具有製造執行系統。依照本發明一較佳實施例，此定義製造執行系統之界面的方法至少包括：提供介面定義檔和XML標示設定檔於該伺服器和該客戶端上，其中介面定義檔係用來執行製造執行系統的複數個服務物件，而XML標示設定檔係使用XML來描述服務物件所具有的介面；執行介面定義檔與XML標示設定檔，以產生XML輸出檔，其中XML標示設定檔係用來做為介面定義檔中所描述之服務物件的至少一輸入參數；提供XML Schema檔於網路伺服器上；以及執行XML Schema檔，以驗證XML輸出檔的內容。

因此，應用本發明，可減少IDL檔的數目至1個，並簡化IDL檔的內容；去除介面儲存裝置；使製造執行系統之資料的格式成為有組織的標準化格式；適用於多種通訊規範；以及提供驗證製造執行系統之資料的功能。

#### 【實施方式】

本發明之定義製造執行系統之界面的系統與方法係使用XML來簡化SiView製造執行系統所使用的IDL檔，並去除介面儲存裝置。

XML係一世界標準語言，且為許多資訊科技(Information Technology；IT)製造商所支援，如美國國際商用機器公司、美國微軟公司(Microsoft)和美國昇陽公司(Sun)等。



##### 五、發明說明 (5)

XML可以當各種領域之描述語言的中介者，即來源描述可以  
先轉換成XML檔，再由目的端將此XML檔轉換成目的端的描  
述語言。XML係一強而有力的資訊交換工具，各個不同領域  
的特殊文件格式，皆可用XML加以定義，來達到網路上文件  
統一的目的。另外，XML具有優良的驗證機制，藉由XML  
Schema檔中的規定，便可自動驗證所存取之資料的內容。  
因此，本發明的主要特徵便是應用XML來簡化SiView製造執  
行系統所使用的IDL檔，並進一步地去除其中之介面儲存裝  
置。由於簡化後之IDL檔只剩下1個，加上1個以XML撰寫的  
XML標示設定檔，而XML標示設定檔的格式具有高度的組織  
性，故可大幅地減少管理檔案之編碼與部署的經常費用。  
請參照第2圖，其繪示本發明之定義製造執行系統之介面的  
系統架構示意圖。本發明之定義製造執行系統之介面的系  
統係以一個介面定義檔202和一個XML標示設定檔204，來取  
代習知系統之複數個IDL檔和複數個介面儲存裝置。其中介  
面定義檔202係用以執行製造執行系統的複數個服務物件  
(Service Objects)。此製造執行系統可為如上所述之  
SiView製造執行系統或其他適當的製造執行系統。而XML標  
示設定檔204係使用XML來描述服務物件所具有的介面。當  
執行介面定義檔202和XML標示設定檔204，來產生XML輸出  
檔206時，XML標示設定檔204係用來做為介面定義檔202中  
所描述之服務物件的至少一輸入參數。當XML輸出檔206在  
伺服器200和客戶端220傳送時，可透過建立在網路伺服器  
230上之XML Schema檔232，來驗證XML輸出檔206的內容。



## 五、發明說明 (6)

由於本發明使用XML之故，伺服器200和客戶端220均需以XML為基礎。

值得一提的是，在使用XML重新描述執行製造執行系統的服務物件之後，基本上已不再需要CORBA中的IDL檔，故執行製造執行系統之伺服器200和客戶端220的通訊規範210不一定只能建立在CORBA上，通訊規範210亦可為其他通訊基礎架構，例如：純粹的傳輸控制/網路通訊協定

(Transmission Control Protocol/Internet Protocol ; TCP/IP)、TIBCO Rendezvous(RV)或單物件存取協定(Simple Object Access Protocol ; SOAP)等。

以下以一實施例來說明本發明：

請參照第3圖，其顯示本發明之較佳實施例之介面定義檔的內容示意圖。如第3圖所示，本發明之介面定義檔202相當簡潔，只需描述一個稱為SiView\_Transection()的服務物件，此服務物件只有一個ASCII型式的輸入參數，而其輸出與輸入均是字串(String)型式，因而可以XML的格式來描述。

請參照第4圖，其顯示本發明之較佳實施例之XML標示設定檔的示意內容示意圖。XML標示設定檔204係定義SiView\_Transection()的輸入參數XML-Argv (如第3圖所示)，XML-Argv係用來查明批貨(Lot)的資訊。XML標示設定檔204並設定有使用XML Schema檔(如TestSchema.xml)來驗證執行後的輸出內容。此驗證動作可幫助系統人員進行版本(Version)維護，避免不同版本的檔案造成困擾。



#### 五、發明說明 (7)

至於XML輸出檔和XML Schema檔的內容，請參照第5圖和第6圖，第5圖為顯示本發明之較佳實施例之XML輸出檔的內容示意圖；第6圖為顯示本發明之較佳實施例之XML Schema檔的內容。執行SiView\_Transaction()後，產生XML輸出檔206，其中設定有使用XML Schema檔232(即TestSchema.xml)。

綜上所述，本發明之定義製造執行系統之介面之方法的流程說明如下：

請參照第7圖，其繪示本發明之定義製造執行系統之介面之方法的流程示意圖。首先，進行步驟300，以提供介面定義檔和XML標示設定檔於伺服器 and 客戶端上，其中介面定義檔描述有執行製造執行系統的服務物件，而XML標示設定檔係使用XML來描述服務物件所具有的介面。接著，執行介面定義檔與XML標示設定檔而產生XML輸出檔(步驟310)，其中XML標示設定檔係用來做為介面定義檔中所描述之服務物件的至少一個輸入參數。接著，進行步驟320，以提供XML Schema檔於網路伺服器上。然後，執行XML Schema檔，以驗證XML輸出檔的內容(步驟330)。

由上述本發明較佳實施例可知，應用本發明之定義製造執行系統之介面的系統與方法的優點為：由於減少IDL檔的數目至1個並簡化IDL檔的內容，且去除介面儲存裝置，故大幅地減少管理檔案之編碼與部署的經常費用；由於使製造執行系統之資料的格式成為有組織的標準化格式，並提供驗證製造執行系統之資料的功能，故增加知識管理的效



五、發明說明 (8)

率；以及可適用於多種通訊規範，毋須限定於CORBA，故具有高度的可移植性。

雖然本發明已以一較佳實施例揭露如上，然其並非用以限定本發明，任何熟習此技藝者，在不脫離本發明之精神和範圍內，當可作各種之更動與潤飾，因此本發明之保護範圍當視後附之申請專利範圍所界定者為準



#### 圖式簡單說明

第1圖為繪示習知之定義製造執行系統之介面的系統架構示意圖。

第2圖為繪示本發明之定義製造執行系統之介面的系統架構示意圖。

第3圖為顯示本發明之較佳實施例之介面定義檔的內容示意圖。

第4圖為顯示本發明之較佳實施例之XML標示設定檔的示意內容示意圖。

第5圖為顯示本發明之較佳實施例之XML輸出檔的內容示意圖。

第6圖為顯示本發明之較佳實施例之XML Schema檔的內容。

第7圖為繪示本發明之定義製造執行系統之介面之方法的流程示意圖。

#### 【元件代表符號簡單說明】

100：伺服器

102：IDL檔

104：IFR

110：CORBA

120：客戶端

200：伺服器

202：介面定義檔

204：XML標示設定檔



圖式簡單說明

- 206 : XML 輸出檔
- 210 : 通訊規範
- 220 : 客戶端
- 230 : 網路伺服器
- 232 : XML Schema 檔
- 300 : 提供介面定義檔和XML標示設定檔於伺服器和客戶端上
- 310 : 執行介面定義檔與XML標示設定檔以產生XML輸出檔
- 320 : 提供XML Schema 檔於網路伺服器上
- 330 : 執行XML Schema 檔以驗證XML輸出檔





## 六、申請專利範圍

1. 一種定義製造執行系統(Manufacturing Execution System ; MES)之介面(Interface)的系統，藉以處理一伺服器(Server)和一客戶(Client)端間的互動(Transaction)，該伺服器和該客戶端係以可延伸標記語言(Extensible Markup Language Base ; XML Base)為基礎(Base)，該伺服器和該客戶端間的互動係建構在一通訊規範上，該伺服器具有一製造執行系統，該定義製造執行系統之介面的系統至少包括：

一介面定義檔，用以執行該製造執行系統的複數個服務物件(Service Objects)；

一XML標示設定(Tag Set)檔，其中該XML標示設定檔係使用XML來描述該些服務物件所具有的介面；以及

一XML Schema(綱要)檔，其中該XML Schema檔係建立在一網路伺服器(Web server)上，用以驗證(Validate)由執行該介面定義檔和該XML標示設定檔而得之一XML輸出檔的內容，其中該XML標示設定檔係用來做為該介面定義檔中所描述之該些服務物件的至少一輸入參數。

2. 如申請專利範圍第1項所述之定義製造執行系統之介面的系統，其中該製造執行系統為由美國國際商用機器公司(IBM)所製造的SiView製造執行系統。

3. 如申請專利範圍第1項所述之定義製造執行系統之介面的



#### 六、申請專利範圍

系統，其中該通訊規範為共用物件請求仲介者架構(Common Object Request Broker Architecture ; CORBA)。

4. 如申請專利範圍第1項所述之定義製造執行系統之介面的系統，其中該通訊規範為純粹的傳輸控制/網路通訊協定(Transmission Control Protocol/Internet Protocol ; TCP/IP)。

5. 如申請專利範圍第1項所述之定義製造執行系統之介面的系統，其中該通訊規範為TIBCO Rendezvous(RV)。

6. 如申請專利範圍第1項所述之定義製造執行系統之介面的系統，其中該通訊規範為單物件存取協定(Simple Object Access Protocol ; SOAP)。

7. 一種定義製造執行系統之介面的系統，藉以處理一伺服器和一客戶端間的互動，該伺服器和該客戶端間的互動係建構在CORBA上，該伺服器和該客戶端係以XML為基礎，該伺服器具有由美國國際商用機器公司所製造的一SiView製造執行系統，該定義製造執行系統之介面的系統至少包括：

一介面定義檔，用以執行該SiView製造執行系統的複數個服務物件；

一XML標示設定檔，其中該XML標示設定檔係使用XML來描述



## 六、申請專利範圍

該些服務物件所具有的介面；以及

一XML Schema檔，其中該XML Schema檔係建立在一網路伺服器上，用以驗證由執行該介面定義檔和該XML標示設定檔而得之一XML輸出檔的內容，其中該XML標示設定檔係用來做為該介面定義檔中所描述之該些服務物件的至少一輸入參數。

8. 一種定義製造執行系統之介面的方法，藉以處理一伺服器和一客戶端間的互動，該伺服器和該客戶端間的互動係建構在一通訊規範上，該伺服器和該客戶端係以XML為基礎，該伺服器具有一製造執行系統，該定義製造執行系統之介面的方法至少包括：

提供一介面定義檔和一XML標示設定檔於該伺服器和該客戶端上，其中該介面定義檔係用來執行該製造執行系統的複數個服務物件，而該XML標示設定檔係使用XML來描述該些服務物件所具有的介面；

執行該介面定義檔與該XML標示設定檔，以產生一XML輸出檔，其中該XML標示設定檔係用來做為該介面定義檔中所描述之該些服務物件的至少一輸入參數；

提供一XML Schema檔於一網路伺服器上；以及

執行該XML Schema檔，以驗證該XML輸出檔的內容。

9. 如申請專利範圍第8項所述之定義製造執行系統之介面的方法，其中該製造執行系統為由美國國際商用機器公司所



六、申請專利範圍

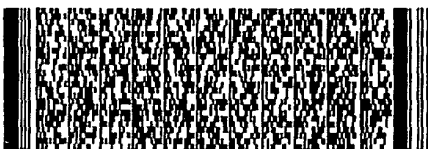
製造的SiView製造執行系統。

10. 如申請專利範圍第8項所述之定義製造執行系統之界面的方法，其中該通訊規範為CORBA。

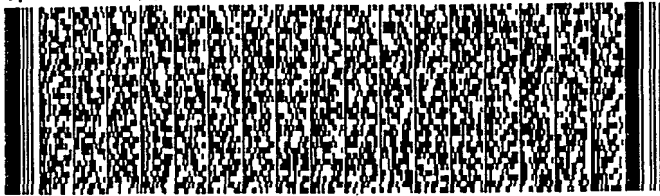
11. 如申請專利範圍第8項所述之定義製造執行系統之界面的方法，其中該通訊規範為純粹的TCP/IP。

12. 如申請專利範圍第8項所述之定義製造執行系統之界面的方法，其中該通訊規範為TIBCO RV。

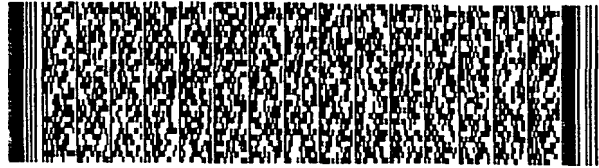
13. 如申請專利範圍第8項所述之定義製造執行系統之界面的方法，其中該通訊規範為SOAP。



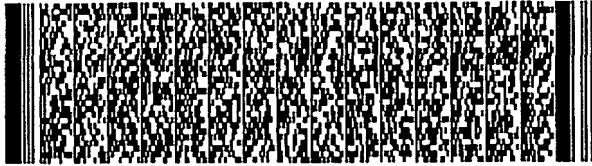
第 1/18 頁



第 2/18 頁



第 2/18 頁



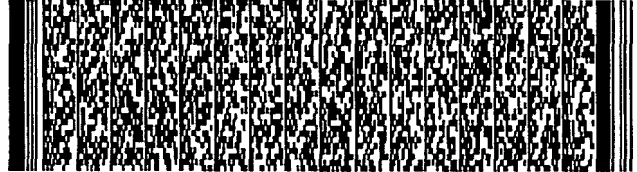
第 3/18 頁



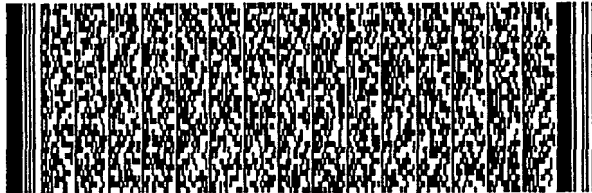
第 4/18 頁



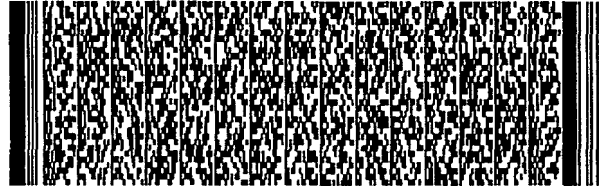
第 5/18 頁



第 5/18 頁



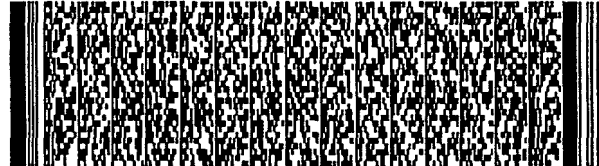
第 6/18 頁



第 6/18 頁



第 7/18 頁



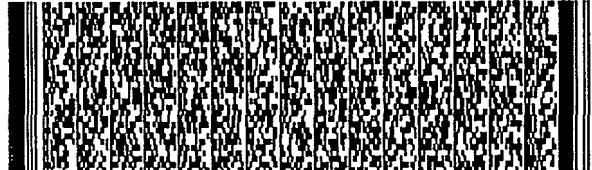
第 7/18 頁



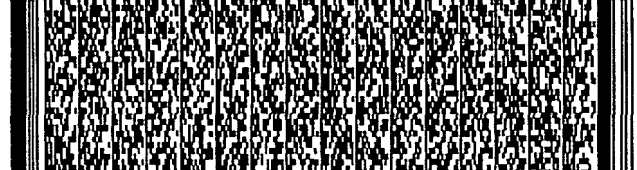
第 8/18 頁



第 8/18 頁



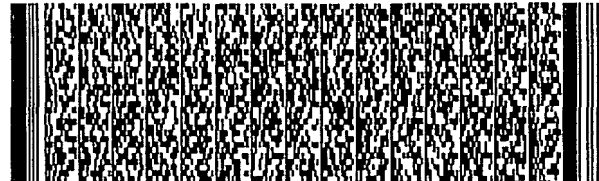
第 9/18 頁



第 9/18 頁



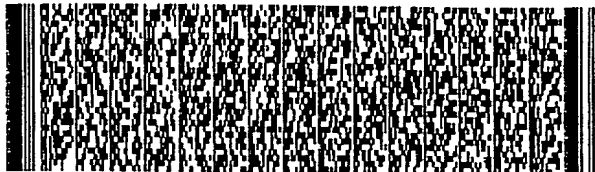
第 10/18 頁



第 10/18 頁



第 11/18 頁



第 11/18 頁



第 12/18 頁



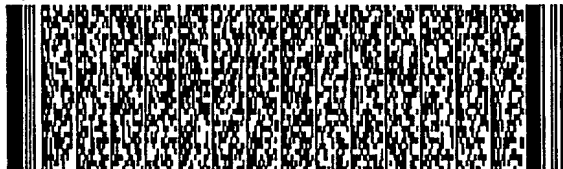
第 13/18 頁



第 14/18 頁



第 15/18 頁



第 15/18 頁



第 16/18 頁



第 16/18 頁



第 17/18 頁

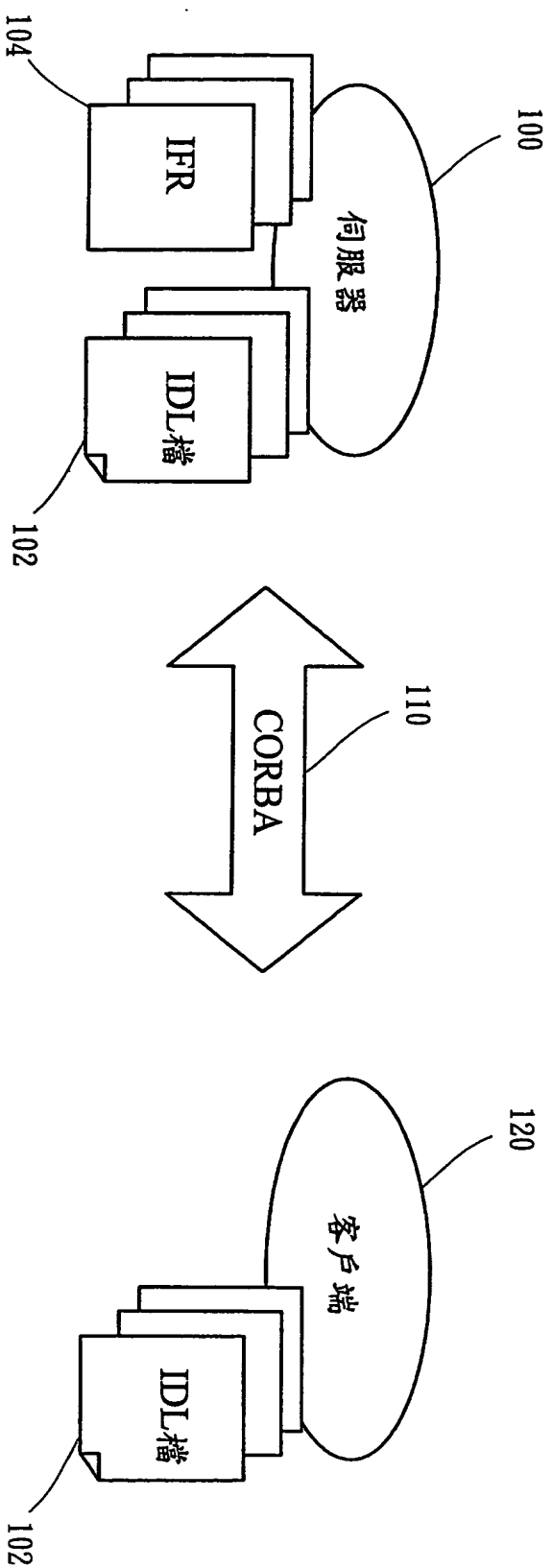


第 17/18 頁

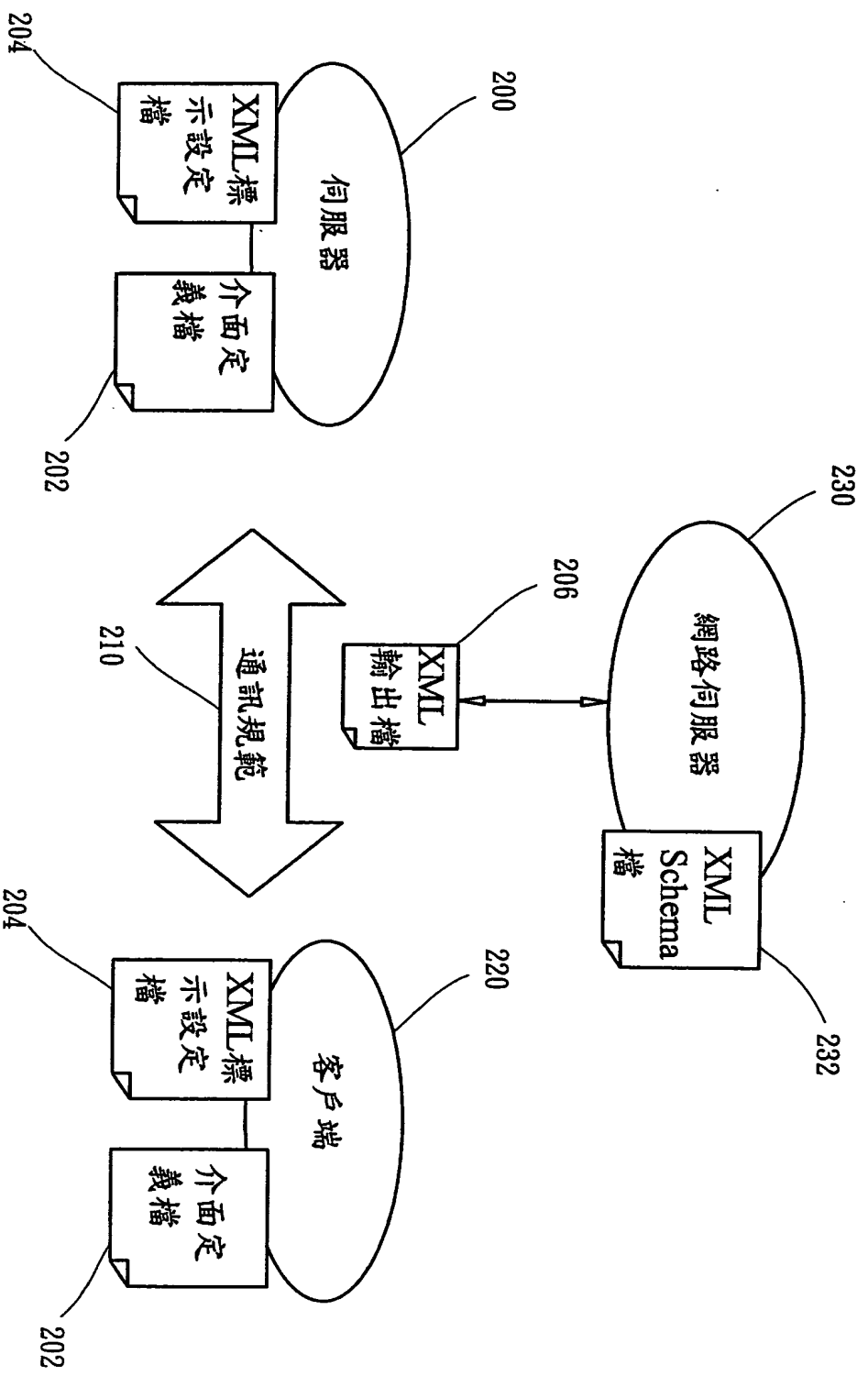


第 18/18 頁



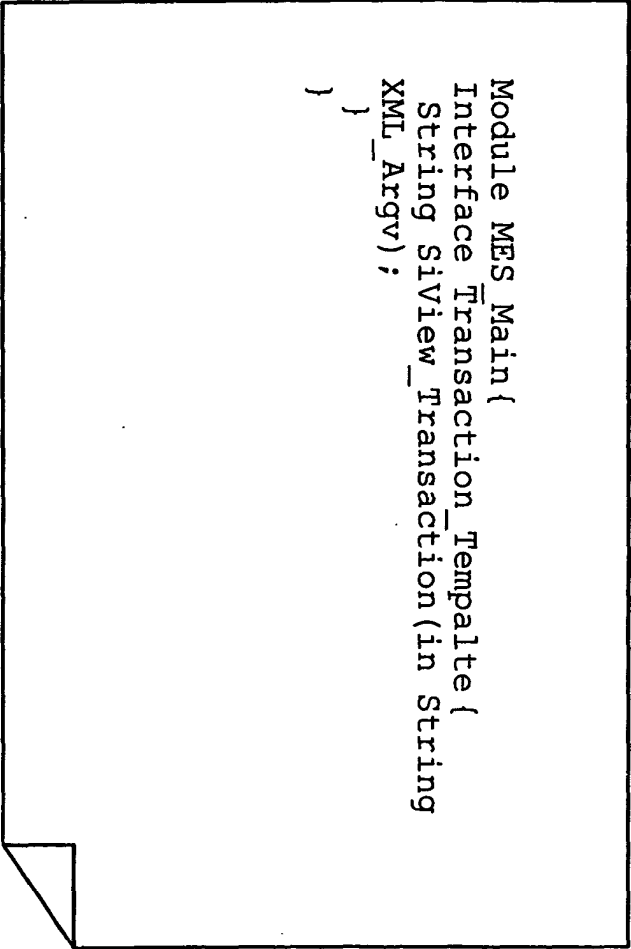


第 1 圖



第 2 圖





```
Module MES_Main{  
  Interface Transaction_Template{  
    String Siview_Transaction(in String  
      XML_Argv);  
  }  
}
```

第 3 圖

```
<?xml version="1.0" ?>
<Transactions xmlns="x-schema:http://SomeWebServer/TestsSchema.xml">
<Transaction ID="LotInfoInq" Action="Inquiry" FuncID="0001">
<Header sno="00100">
<From node="MyPC" sys="OMT" />
<User pwd="123">ABC</User>
</Header>
<Contents>
<Lots>
<Lot lot_id="ABC100.00">
<Argv1 txt="1" />
<Argv2 txt="2" />
</Lot>
</Lots>
</Contents>
</Transaction>
</Transactions>
```

```
<?xml version="1.0" ?>
<Transactions xmlns="x-schema:http://SomeWebServer/TestSchema.xml?"
<Transaction ID="LotInfoInqReply" Action="ResultReply" FuncID="0005">
<Header sno="00101">
  <From node="MessServer" sys="MMS" />
  <Msg rc="0" msg_id="0001">Normal End</Msg>
</Header>
<Contents>
<Lots>
  <Lot lot_id="ABC100.00">
    <HoldState txt="NotOnHold"/>
    <Attr2 txt="reply stuff..." />
    <Attr3 txt="reply stuff..." />
  </Lot>
</Lots>
</Contents>
</Transaction>
</Transactions>
```

```
<Schema xmlns="urn:schemas-microsoft-com:xml-data"
  xmlns:dt="urn:schemas-microsoft-com:datatypes">
  <ElementType name="Attr2" content="elOnly">
    </ElementType>
  <ElementType name="Attr3" content="elOnly">
    </ElementType>
  <AttributeType name="txt" required="yes" dt:type="string"/>
  <ElementType name="HoldState" content="elOnly">
    <attribute type="txt"/>
  </ElementType>
  <AttributeType name="Lot_id" required="yes" dt:type="string"/>
  <ElementType name="Lot" content="mixed">
    <attribute type="Lot_id"/>
    <element type="HoldState"/>
    <element type="Attr2"/>
    <element type="Attr3"/>
  </ElementType>
```

## 第 6 圖

```
<ElementType name="Lots" content="eltonly">
  <element type="Lot"/>
</ElementType>

<ElementType name="Contents" content="eltonly">
  <element type="Lots"/>
</ElementType>

<AttributeType name="rc" required="yes" dt:type="string"/>
<AttributeType name="msg_id" required="yes" dt:type="string"/>
<ElementType name="Msg" content="eltonly">
  <attribute type="rc"/>
  <attribute type="msg_id"/>
</ElementType>

<AttributeType name="node" required="yes" dt:type="string"/>
<AttributeType name="sys" required="yes" dt:type="string"/>
<ElementType name="From" content="eltonly">
  <attribute type="node"/>
  <attribute type="sys"/>
</ElementType>
```

## 第 6 圖 (續-1)

232

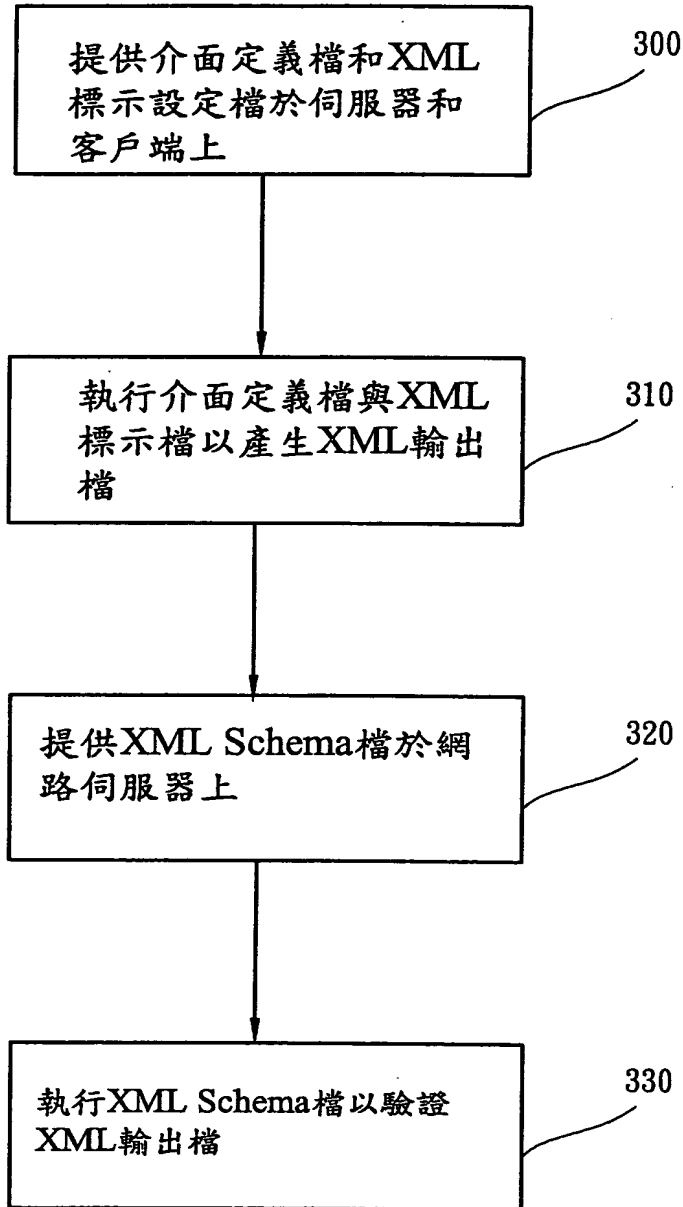
```
<AttributeType name="sno" required="yes" dt:type="string"/>
<ElementType name="Header" content="mixed">
  <attribute type="sno"/>
  <element type="From"/>
  <element type="Msg"/>
</ElementType>
```

```
<AttributeType name="ID" required="yes" dt:type="string"/>
<AttributeType name="Action" dt:type="string" required="yes"/>
<AttributeType name="FuncID" dt:type="string" required="yes"/>
<ElementType name="Transaction" content="mixed">
  <attribute type="ID"/>
  <attribute type="Action"/>
  <attribute type="FuncID"/>
  <element type="Header"/>
  <element type="Contents"/>
</ElementType>
```

```
<ElementType name="Transactions" content="elOnly">
  <element type="Transaction"/>
</ElementType>

</Schema>
```

## 第 6 圖 (續-2)



第 7 圖